

Extension of OpenMC for Fixed Source Transmutation Calculations

Computational Science Division

About Argonne National Laboratory

Argonne is a U.S. Department of Energy laboratory managed by UChicago Argonne, LLC under contract DE-AC02-06CH11357. The Laboratory's main facility is outside Chicago, at 9700 South Cass Avenue, Lemont, Illinois 60439. For information about Argonne and its pioneering science and technology programs, see www.anl.gov.

DOCUMENT AVAILABILITY

Online Access: U.S. Department of Energy (DOE) reports produced after 1991 and a growing number of pre-1991 documents are available free at OSTI.GOV (<http://www.osti.gov/>), a service of the U.S. Dept. of Energy's Office of Scientific and Technical Information

Reports not in digital format may be purchased by the public from the National Technical Information Service (NTIS):

U.S. Department of Commerce
National Technical Information Service
5301 Shawnee Rd
Alexandria, VA 22312
www.ntis.gov
Phone: (800) 553-NTIS (6847) or (703)
605-6000 Fax: (703) 605-6900
Email: **orders@ntis.gov**

Reports not in digital format are available to DOE and DOE contractors from the Office of Scientific and Technical Information (OSTI):

U.S. Department of Energy
Office of Scientific and Technical Information
PO. Box 62
Oak Ridge, TN 37831-0062
www.osti.gov
Phone: (865) 576-8401
Fax: (865) 576-5728
Email: **reports@osti.gov**

Disclaimer

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor UChicago Argonne, LLC, nor any of their employees or officers, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of document authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof, Argonne National Laboratory, or UChicago Argonne, LLC.

Extension of OpenMC for Fixed Source Transmutation Calculations

prepared by
Paul K. Romano

Computational Science Division

October 1, 2020

1 Project Summary

This report documents work performed under a Strategic Partnership Project between Argonne National Laboratory (ANL) and the United Kingdom Atomic Energy Authority (UKAEA). The overall goal of this project is to extend OpenMC [1], a community-developed Monte Carlo particle transport code, to be able to perform fixed-source transmutation calculations. In fission and fusion reactors, the high flux of energetic neutrons causes materials within the reactor to “transmute,” or undergo a nuclear reaction that results in the addition/removal of neutrons and protons from the nucleus of an atom. If subjected to these reactions for long enough, the overall composition and physical properties of the material itself begin to change as a result of transmutation. Such a feature is vital for predicting the decrease in tritium production rate within a breeder blanket during the lifetime of a fusion reactor.

OpenMC is capable of simulating neutron transport in fission/fusion systems, thereby allowing it to estimate the flux that causes transmutation. It is also capable of solving the transmutation equations, which determine how the composition of a material changes over time due to neutron irradiation and radioactive decay. However, solving the transmutation equations was previously only possible when the source of neutrons came from a fission system. In a fusion system, the source of neutrons is typically determined by a separate code and then given as an input to the particle transport simulation. This is known as a fixed source calculation. Through this project, we have extended OpenMC to solve the transmutation equations for a fixed source calculation.

Evaluating the change in material compositions due to transmutation and its effect on physical properties is of key importance to a range of engineering analyses for fission and fusion systems. For example, in a fusion reactor, estimating the dose rate at different physical locations resulting from irradiated materials in the reactor allows designers to ensure that workers are not exposed to doses beyond applicable regulations. In order to properly dispose of irradiated materials, designers also need to estimate the radiotoxicity, which again relies on knowledge of the material composition at some future time.

The specific tasks for this project that were agreed to between ANL and UKAEA were as follows:

1. Make changes and additions in the `openmc.deplete` and related modules in OpenMC to support transmutation calculations following a fixed source transport simulation.
2. Make necessary changes to OpenMC to model transmutation due to an arbitrary set of reactions needed for fusion applications. Use this new capability to generate a depletion chain file based on the TENDL nuclear data library.
3. Improve the `openmc.deplete` module in OpenMC to keep track of gases produced as a by-product of nuclear reactions during transmutation calculations.
4. Validate the new capabilities by carrying out fixed-source transmutation calculations on a suitable benchmark problem using OpenMC and a comparable Monte Carlo neutron transport code.

The following sections describe work performed toward each of these tasks.

2 Work Performed

2.1 Task 1: Fixed Source Transmutation

Prior to this work, the `openmc.deplete` module was only designed for transmutation calculations for fission reactor systems. From a practical standpoint, this affects how tally results are normalized. In order to construct the depletion matrix, absolute reaction rates are needed. However, tally results in OpenMC (and most other Monte Carlo codes) are given per source particle. Thus, the results need to be normalized based on the actual source rate in order to determine an absolute reaction rate that is used when constructing the depletion matrix. In a fission system, the source rate is determined by looking at the energy deposited during a simulation and comparing it to the user-specified power. The ratio of these quantities determines a normalization factor that is applied to tally results. For a fixed source problem, the source rate is known *a priori*, so the normalization is straightforward—the tally results, which are given per source particle, just need to be multiplied by the known source rate.

Two user-facing changes were made in order to accommodate the specification of a source rate for fixed source transmutation calculations. First, the `Integrator` classes, which control the algorithm for advancing time in the system, were modified to take a new `source_rates` argument that can be used in lieu of the `power` or `power_density` arguments. The length of `source_rates` should match the number of timesteps provided, or if a single value is given it is assumed to apply to all timesteps. Additionally, a zero source rate for a given timestep will result in a decay-only calculation where all reaction rates are zero. The second change needed is in the `Operator` class, which has an argument called `normalization_mode` that indicates how tally results should be normalized. A new option was added for this argument ("source-rate") that tells the transport operator to normalize tally results by the source rate that is provided.

Internally, many function interfaces needed slight modification. Prior to this addition, a power value in Watts was passed through a series of function calls from the depletion integrator to the transport operator. The argument for these functions was renamed from `power` to `source_rate` and now represents either a power in Watts or a source rate in neutrons per second.

Several other small changes were made in order to get a basic fixed source transmutation capability working:

- The construction of depletion chains was improved by adding explicit methods on `openmc.deplete.Nuclide` for adding reactions and decay modes. This makes it easy to build a simple depletion chain with only a few reactions or decay modes for testing.
- Before, the depletion module accessed internal tally results directly, which are not normalized by the number of tally realizations. This actually works in k -eigenvalue depletion calculations because the absolute tally results are just renormalized based on the power, but in the case of fixed-source transmutation calculations, the properly batch-normalized values are needed. Thus, all uses of unnormalized tally results now rely on the mean value, which accounts for the number of realizations.

Version 0.12 of OpenMC [2] allows a user to specify a timestep in depletion with zero power, resulting in a period of time during which only radioactive decay takes place. However, a

transport solve was still called even though it was unnecessary. OpenMC has now been modified to check for a zero power or source rate and automatically return zero reaction rates for all relevant reactions without running a transport solve. A separate issue is that a transport solve was always initiated at the end of the last depletion timestep (mostly to determine a final k_{eff} value). However, for activation calculations, this transport solve serves no purpose. An argument was added on the `Integrator.integrate` method to indicate whether a transport solve at the end of the last timestep should be performed, thereby giving the user an option to avoid this transport solve.

Relevant Code Changes

- <https://github.com/openmc-dev/openmc/pull/1628>
- <https://github.com/openmc-dev/openmc/pull/1636>

2.2 Task 2: Extending Transmutation Reactions

Version 0.12 of OpenMC [2] defaults to using the following reactions when creating a depletion chain: $(n,2n)$, $(n,3n)$, $(n,4n)$, (n,γ) , (n,p) , (n,α) , and fission. These reactions are the ones that have traditionally been included for reactor depletion calculations. The reactions included in the chain were controlled by a global variable in `openmc/deplete/chain.py`. This variable was expanded to include all possible transmutation reactions allowed by the ENDF-6 format, but is no longer used as the direct means of indicating what reactions should be included in the chain. Instead, the `Chain.from_endf` classmethod was modified to include a new argument, `reactions`, that accepts an iterable of reaction names indicating what reactions should be included in the depletion chain. This allows a user to override the default reactions that would normally be included.

Relevant Code Changes

- <https://github.com/openmc-dev/openmc/pull/1618>

2.2.1 TENDL Depletion Chain

A script was written to generate a depletion chain based on TENDL-2019 incident neutron sublibrary files. Because TENDL does not have a decay or fission product yield (FPY) sublibrary, another library must be used to provide decay and FPY data. JEFF 3.3 was used to provide this data. The script relies on existing capabilities implemented in the `Chain.from_xml` method. Using this method on the TENDL-2019 incident neutron and JEFF-3.3 decay and FPY sublibraries produces a chain with 3,851 nuclides. The `Chain.reduce` method can reduce the total number of nuclides to 3,117 by eliminating nuclides that do not appear as reaction or decay products based on a starting set of all stable and very long-lived nuclides.

The generation of this chain has identified a number of issues. OpenMC's `deplete` module currently adds all nuclides with reaction data to a material when performing a depletion calculation. For a library like TENDL-2019, this will result in a severe slowdown in the calculation

time because of the extra cross-section lookups necessary for each nuclide as a particle travels through a “depletable” material. There are a few possible ways around this:

- Tallying a multigroup flux and using that to calculate reaction rates by collapsing with the continuous-energy cross sections
- Having an option to not add nuclides that don’t originally appear in a material and set their reaction rates to zero
- Building a “simplified” chain that includes a limited set of nuclides

The other issue with the TENDL chain is that for fusion applications, one generally does care about all possible transmutation reactions. In the S30 (default) version of TENDL-2019, 36 separate reactions can result in transmutation. This is in contrast to fission applications where typically the number of transmutation reactions is limited to six or seven.

Relevant Code Changes

- <https://github.com/openmc-dev/data/pull/41>
- <https://github.com/openmc-dev/openmc/pull/1629>

2.2.2 Flux-based reaction rates

Calculating reaction rates by collapsing a multigroup flux spectrum with continuous-energy cross sections [3] has several advantages (and disadvantages) relative to directly tallying reaction rates. First, tallying the flux instead of reaction rates scales better with an increasing number of transmutation reactions; as pointed out above, for fusion applications, over 30 reaction channels may need to be accounted for. Additionally, tallying the flux does not require introducing all nuclides of interest into a material at arbitrarily low densities, as is currently done in `openmc.deplete`. Together, these two advantages of flux-based tallies can lead to significant performance improvements for transmutation analyses. However, one significant downside of the flux method is that the number of energy groups required to obtain sufficient accuracy on the collapsed reaction rates may be very high. In this case, the memory requirement of the flux tally can become quite prohibitive if there are many depletable materials in the model. One other downside of the flux method is that it is generally harder to account for temperature dependence of materials in a simulation. When tallying reaction rates directly, one normally has the cross section evaluated exactly at the temperature needed (or interpolated between two bounding temperatures). When collapsing the reaction rate afterward, it is more complicated to account for temperature.

The `openmc.deplete` module has been extended to allow reaction rates to be calculated either via the traditional method (tallying reaction rates directly) or by tallying the flux spectrum and collapsing one-group reaction rates after the transport calculation has completed. The `Operator` class already relies on a reaction rate interface through the abstract `ReactionRateHelper` class. A new subclass of this abstract class called `FluxCollapseHelper` was implemented that automatically creates flux tallies in any depletable material using a user-defined energy group structure. Additionally, a `Nuclide::collapse_rate` method was added

at the C++ layer that accepts a reaction MT number, temperature, and energies/fluxes and returns a collapsed one-group reaction rate. Python bindings to the `Nuclide::collapse_rate` were written so that the underlying C++ method can be called via the `openmc.lib` Python module. Finally, two new arguments were added on `Operator: reaction_rate_mode` and `reaction_rate_opts`. The former argument specifies what method is to be used for calculating reaction rates ("direct" or "flux"), and the latter argument can be used to specify energy group boundaries for tallying the flux.

The implementation of `Nuclide::collapse_rate` relies on a lower-level method on the `Reaction` C++ class that actually performs the integration of the flux multiplied by the reaction cross section. When the temperature of a material is not exactly equal to one of the temperatures at which the reaction cross section is tabulated at, the reaction rate is calculated at the bounding temperatures and then interpolated accordingly. This approximation is equivalent to interpolating the cross sections in temperature and then integrating over energy and should be reasonably accurate provided a sufficiently fine temperature discretization. We also note that in high energy problems, the temperature has a very small impact on the neutron cross sections and hence the reaction rates.

Relevant Code Changes

- <https://github.com/openmc-dev/openmc/pull/1660>

2.3 Task 3: Tracking Light Nuclide Production

Prior to this work, when OpenMC built a transmutation matrix, it did not account for the production of light nuclides as a by-product of nuclear reactions. For example, the (n,p) reaction produces one atom of ^1H .

In `chain.py`, a dictionary was created that stores a mapping from reaction names to tuples of reaction information: ENDF MT numbers, the change in mass number and atomic number resulting from the reaction, and a list of light nuclides produced. Then in `Chain.form_matrix`, this information is used to add contributions to the corresponding light nuclide terms when constructing the transmutation matrix.

Relevant Code Changes

- <https://github.com/openmc-dev/openmc/pull/1555>

2.4 Task 4: Testing and Benchmarking

2.4.1 Single nuclide irradiation test

If we consider the irradiation of a single nuclide, the solution for the nuclide density after an irradiation time of τ can be expressed analytically. Assuming the nuclide is stable and is subject to a flux, ϕ , the rate of change of the nuclide density is

$$\frac{dn}{dt} = -f\sigma\phi n(t) \quad (1)$$

where $n(t)$ is the nuclide density, f is the source rate, and σ is the transmutation cross section. The solution to Eq. (1) is

$$n(\tau) = n_0 \exp(-f \sigma \phi \tau) \quad (2)$$

where $n_0 = n(0)$. The transmutation reaction rate at time zero is $r = \sigma \phi n_0$, so that

$$n(\tau) = n_0 \exp\left(-\frac{f r \tau}{n_0}\right). \quad (3)$$

We can then evaluate Eq. (3) for any f and τ by running an initial transport calculation to determine the reaction rate, r . Our test problem is a 10 cm sphere of tungsten at 19.3 g/cm³ with a 1 MeV neutron source at the center of the sphere. Although this model contains the five naturally occurring isotopes of tungsten, ¹⁸⁶W is effectively isolated because there is no possibility of production from the lighter isotopes (¹⁸⁵W does not have incident neutron data in most data libraries). Thus, ¹⁸⁶W is only depleted by the neutron flux and is not produced otherwise. At neutron energies of 1 MeV and below, the transmutation is driven almost entirely by the (n, γ) reaction, so obtaining the (n, γ) reaction rate allows us to evaluate Eq. (3) for ¹⁸⁶W.

A simulation of the tungsten sphere model was run using ENDF/B-VII.1 data with a source rate of 1.0 and produced a capture rate of 0.02366 reactions per source neutron. To reduce the density of ¹⁸⁶W by a factor of x , we would need

$$\frac{n(\tau)}{n_0} = \exp\left(-\frac{f r \tau}{n_0}\right) = \frac{1}{x}. \quad (4)$$

Solving for f provides the source rate necessary to achieve a factor of x reduction in density:

$$f = \frac{n_0}{r \tau} \ln x. \quad (5)$$

Based on the observed capture rate, Eq. (5) was used to estimate the source rate needed to reduce the density by a factor of two and four for various times, τ . Fixed source transmutation calculations were then run with these source rates to see what the observed density would be at the end of the calculation. Table 1 shows the observed ratio of $n(\tau)$ to n_0 , demonstrating that it closely matches the prediction.

Table 1: Single nuclide problem results

Source rate [neutron/s]	τ [d]	$n(\tau)/n_0$ Expected	$n(\tau)/n_0$ Observed
2.552×10^{22}	1	0.5	0.4999
1.276×10^{22}	2	0.5	0.4999
6.381×10^{21}	4	0.5	0.5000
5.105×10^{22}	1	0.25	0.2499
2.552×10^{22}	2	0.25	0.2500
1.276×10^{22}	4	0.25	0.2502

2.4.2 Single nuclide irradiation and decay

This test is identical to the previous test except that following the irradiation for time τ , a 2 day decay timestep is added where no irradiation occurs. In this test, instead of looking at ^{186}W , we will look at ^{187}W , which has a half-life of exactly 24 hours according to the ENDF/B-VII.1 decay sublibrary. During the initial irradiation period, ^{187}W is produced from $^{186}\text{W}(n,\gamma)^{187}\text{W}$ but also undergoes radioactive decay. The system of equations for the density of ^{186}W and ^{187}W is

$$\begin{aligned}\frac{dn_{186}}{dt} &= -rn_{186}(t) \\ \frac{dn_{187}}{dt} &= rn_{186}(t) - \lambda n_{187}(t)\end{aligned}\tag{6}$$

where λ is the decay constant of ^{187}W and, for simplicity, r now represents a reaction rate per atom that accounts for the source rate. The solution for $n_{187}(t)$ can be determined using the integrating factor method and is found to be

$$n_{187}(t) = \frac{rn_0}{\lambda - r} (e^{-rt} - e^{-\lambda t}).\tag{7}$$

It is more convenient to express n_{187} as a fraction of n_0 ,

$$\frac{n_{187}(t)}{n_0} = \frac{r}{\lambda - r} (e^{-rt} - e^{-\lambda t}).\tag{8}$$

After the initial irradiation period, the density of ^{187}W decays exponentially due to radioactive decay. Accounting for this, we can express the solution at any time t as

$$\frac{n_{187}(t)}{n_0} = \begin{cases} \frac{r}{\lambda - r} (e^{-rt} - e^{-\lambda t}) & t \leq \tau \\ \frac{r}{\lambda - r} (e^{-r\tau} - e^{-\lambda\tau}) e^{-\lambda(t-\tau)} & t > \tau \end{cases}\tag{9}$$

Once again, an initial transport calculation to determine the (n,γ) reaction rate in ^{186}W enables us to predict the ^{187}W density relative to n_0 using [Eq. \(9\)](#). Fixed source transmutation calculations were then run with varying source rates and times, all of which reduce the ^{186}W density by a factor of two. [Equation \(9\)](#) was then used to predict the ^{187}W density at time τ . [Table 2](#) shows the observed ^{187}W density relative to n_0 at time τ , as well as the ratio of $n_{187}(\tau + 2)$ to $n_{187}(\tau)$, which is expected to be 0.25 given that ^{187}W has a 1 day half-life. The observed densities match the predicted densities closely.

Table 2: Single nuclide irradiation and decay problem results

Source rate [neutron/s]	τ [d]	$\frac{n_{187}(\tau)}{n_0}$ Expected	$\frac{n_{187}(\tau)}{n_0}$ Observed	$\frac{n_{187}(\tau+2)}{n_{187}(\tau)}$ Observed
5.105×10^{22}	0.5	0.4142	0.4143	0.25
2.552×10^{22}	1	0.3466	0.3466	0.25
1.276×10^{22}	2	0.2500	0.2501	0.25
6.381×10^{21}	4	0.1458	0.1458	0.25
3.190×10^{21}	8	0.0709	0.0709	0.25

2.4.3 Tritium breeder blanket irradiation

In this test, we model a 10 cm sphere of Li_2ZrO_3 at 4.15 g/cm^3 . This material is commonly cited as a possible tritium breeding material for a fusion reactor. The sphere is subject to a pulsed monoenergetic neutron source with an energy of 14 MeV and a source strength of 10^{12} neutrons per second, alternating between 1 day irradiation and 30 day decay steps. A total of four pulses are carried out. For this test, the objective is to look at how much tritium and radioactive isotopes of zirconium (^{93}Zr and ^{95}Zr) are produced. The radioactive isotopes of zirconium present a challenge for material handling; ^{93}Zr is a long-lived isotope and contributes to shutdown dose rate, and ^{95}Zr is a shorter-lived isotope that contributes to decay heat.

Instead of predicting the densities of the isotopes of interest analytically, models were created for both OpenMC and Serpent to enable a cross-code comparison. ENDF/B-VII.1 neutron cross sections and decay/fission product yield data was used in both OpenMC and Serpent for consistency. For OpenMC, a tailored depletion chain was built starting with all naturally occurring isotopes of Li, Zr, and O and following reaction/decay paths several times (using the `Chain.reduce` method). This depletion chain also includes all transmutation reactions; the existing depletion chains for fission applications do not include the (n,t) reaction and would therefore entirely miss the production of tritium from ^6Li .

Fig. 1 shows the atom density of hydrogen isotopes as predicted by OpenMC and Serpent. Excellent agreement in the predicted atom densities is observed. In this test, the decay timesteps are not long enough to observe any appreciable decay in the quantity of ^3H , which has a 12.32 year half-life. Fig. 2 shows the atom density of ^{93}Zr and ^{95}Zr as predicted by OpenMC and Serpent. Once again, excellent agreement is observed. Because ^{95}Zr has a 64 day half-life, there is a $\sim 28\%$ reduction in its atom density over the 30 day decay timesteps.

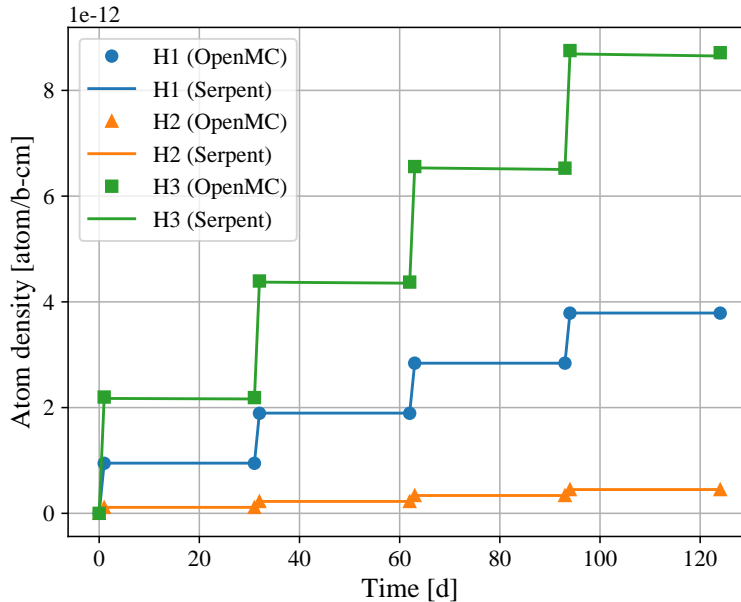


Figure 1: Atom density of hydrogen isotopes in the pulsed tritium breeder blanket problem as predicted by OpenMC and Serpent.

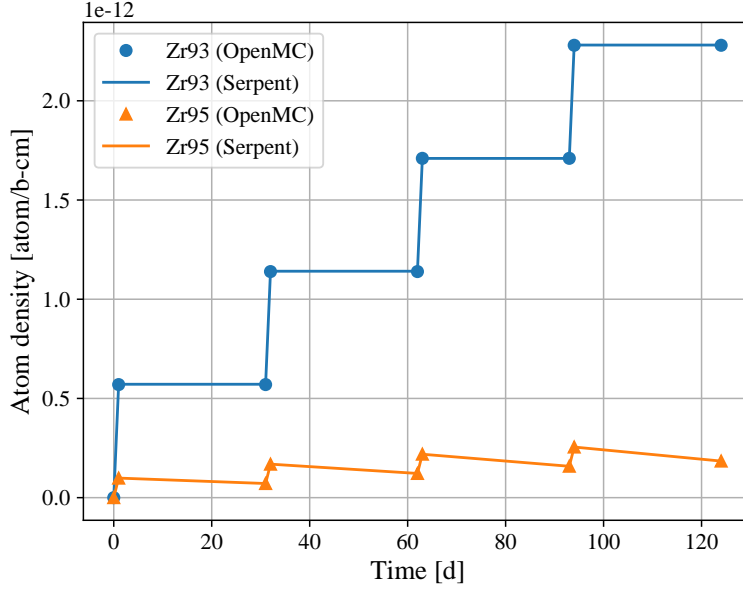


Figure 2: Atom density of zirconium isotopes in the pulsed tritium breeder blanket problem as predicted by OpenMC and Serpent.

3 Conclusions

OpenMC has been extended to support depletion/transmutation calculations in conjunction with a fixed source transport calculation. The classes within the `openmc.deplete` module now support directly specifying a source rate that is used to normalize tallies. Other improvements have been made as well, including the ability to perform a decay timestep without running a transport calculation. Changes were also made so that depletion chains can be generated with an arbitrary, user-specified list of transmutation reactions. These changes were then used to produce a depletion chain based on the TENDL-2019 data library. Construction of the transmutation matrix was also modified to account for the production of light nuclides from reactions such as (n,t) and (n,α) .

An implementation of a method to calculate reaction rates based on a tallied flux spectrum has been added to OpenMC. This method can improve performance when many transmutation reactions are present in a depletion chain. A thorough comparison of performance and accuracy tradeoffs of this approach compared to the traditional approach of directly tallying reaction rates is beyond the scope of this project but is recommended to be studied in the future. In principle, the flux-based reaction rate method can also avoid the need to introduce all nuclides of interest into a material at arbitrarily low density. However, this is not currently done in OpenMC. Reducing the number of nuclides loaded for transport should help to further reduce the time to solution for a transport solve.

Testing of the new fixed source transmutation capability was carried out using three test problems. The first two problems consist of a sphere of tungsten that is irradiated. The predicted atom density of ^{186}W and ^{187}W over time was shown to match an analytical solution. The third test problem consists of a sphere of Li_2ZrO_3 that is subject to a pulsed source of irradiation.

This problem was simulated with both OpenMC and Serpent, and predicted atom densities of hydrogen and zirconium isotopes were shown to closely match between the two codes.

While the capabilities that have been introduced by this project are sufficient for predicting material compositions resulting from neutron irradiation, further development is needed to quantify the photon source term resulting from the decay of radioisotopes in an activated material. This will require utilizing information in ENDF decay sublibraries. Shutdown dose rate calculations for fusion systems [4] also require performing a subsequent photon transport calculation using the decay photon source from activated materials, which will require a method in OpenMC for representing and sampling the spatial distribution of the decay source.

One of the conclusions of the work performed here is that it is generally advantageous to construct a tailored depletion chain for a given application. This was already possible using the `Chain.reduce` method and is now also augmented by methods to directly add reactions/decay modes to a nuclide in a depletion chain. Judicious use of these methods can help to significantly improve performance of the transport solution while still retaining accuracy of the quantities of interest.

Currently, OpenMC does not have a means of directly handling energy-dependent capture branching ratios. The existing depletion chains have been constructed with capture branching ratios based on either a thermal or fast reactor spectrum. For fusion applications, it may be worthwhile to investigate how capture branching ratios would be affected by a fusion neutron spectrum.

4 Acknowledgments

This research was supported by a strategic partnership project with the United Kingdom Atomic Energy Authority. The material was based upon work supported by the U.S. Department of Energy, Office of Science, under contract DE-AC02-06CH11357.

References

- [1] P. K. Romano, N. E. Horelik, B. R. Herman, A. G. Nelson, and B. Forget, “OpenMC: A state-of-the-art Monte Carlo code for research and development,” *Ann. Nucl. Energy*, vol. 82, pp. 90–97, 2015. doi: 10.1016/j.anucene.2014.07.048
- [2] P. K. Romano *et al.*, “OpenMC 0.12.0,” Jul. 2020. doi: 10.5281/zenodo.3956803
- [3] W. Haeck and B. Verboomen, “An optimum approach to Monte Carlo burnup,” *Nucl. Sci. Eng.*, vol. 156, pp. 180–196, 2007. doi: 10.13182/NSE07-A2695
- [4] Y. Chen and U. Fischer, “Rigorous MCNP based shutdown dose rate calculations: computational scheme, verification calculations and application to ITER,” vol. 63–64, pp. 107–114, Dec. 2002. doi: 10.1016/S0920-3796(02)00144-8



Computational Science Division

Argonne National Laboratory

9700 South Cass Avenue, Bldg. 240

Lemont, IL 60439

www.anl.gov



Argonne National Laboratory is a U.S. Department of Energy
laboratory managed by UChicago Argonne, LLC